##16

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Dillard, et al.	)
		)
Serial No.:	08/861,989	)
		)
Filed:	05/22/97	)
		)
Title:	Copy Protection For	)
	Updates Transmitted	)
	Via Internet	)
		)
Group No.:	2764	)
		)
Examiner:	Retta, Y.	)

Supplemental Declaration of Kelly Dillard and David Goddard

TC 2700 MAIL ROOM
AUG -8 2000
RECEIVED

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir,

I, Kelly Dillard, of Olathe, Kansas; and I, David Goddard, of Overland Park, Kansas, declare:

4. Exhibit A of this Supplemental Declaration contains a print out of the first and last page of a source code file for the computer routine "CRYPTDB."

5. Exhibit B of this Supplemental Declaration contains the print out of the first and last page of a source code file for the computer routine "UNCRPTDB."

6. Exhibit C of this Supplemental Declaration is a windows directory listing for the directory containing the files identified in paragraphs 4 and 5 above and shows that the files CRYPTD and UNCRPTDB were last modified on ▮▮▮▮▮▮▮.

7. The source code files identified in paragraphs 4 and 5 above are used to implement one preferred embodiment of the invention.

8.   In ████████ we actually reduced the invention to practice by conducting a proof of concept demonstration of the invention using the code files identified in paragraphs 4 and 5 above.

   The undersigned declares further that all statements made herein of his own knowledge are true and that all
5 statements made on information and belief are believed to be true and further that these statements are made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issuing thereon.

10 Signature: _____
   Post Office Address: 20120 W. 121st Lane
                            Olathe, KS 66061

   Date: 7/26/00
15

   Signature: _____
   Post Office Address: 9608 West 104th Terrace
20   Overland Park, KS 66212

   Date: 7/26/00

25

```
/*+module definition*****************************************************
******
**
** Copyright (c) AlliedSignal Inc.
**
**
** Name       : CRYPTDB
**
** Description : This module is a very crude attempt to encrypt the dat
abase
**               file used by KLX100.EXE.  The encryption has a cumulat
ive
**               randomizing effect on the output data, but is of cours
e
**               not bullet-proof from super-sleuth crypt experts.
**
** Global Procedures:
**   Name                        Type      Abstract
**   ----                        ----      --------
**
** Header for Code Management Software
**
**
***********************************************************************
****-*/
#include <stdio.h>
#include <stdlib.h>
#include <io.h>
#include <stdarg.h>
#define FALSE 0;
#define TRUE 1;
#define MIDX_135         0x80            /* master index location for KL
X 135 */
#define MIDX_90          0x42            /* master index location for KL
N 90 */
#define MIDX_RS_135      14             /* 14 bytes per index entry */
#define MIDX_RS_90       3              /* 3 bytes per index entry for
90 */
#define TAG_SIZE         162            /* 162 total bytes tagged on en
d */
#define KLN90_TYPE 1
#define OTHER_TYPE 2

typedef unsigned char BYTE;

/* KLN 90 database modified memory structure */
typedef struct
```

```
    /* Compute table of CRC's */
    crc_32_tab[0] = 0x00000000L;
    for (i = 1; i < 256; i++)
    {
        c = 0;
        for (k = i | 256; k != 1; k >>= 1)
        {
            c = c & 1 ? (c >> 1) ^ e : c >> 1;
            if (k & 1)
                c ^= e;
        }
        crc_32_tab[i] = c;
    }
}


/*----------------------------------------------------------------
-------*/
/* compute a CRC for a given byte stream
        */
/*----------------------------------------------------------------
-------*/
unsigned long get_crc( void *buffer, register int length )
{
    register unsigned long crcval = 0xffffffffL;
    register BYTE *b = buffer;
    while (length--)
        crcval = crc_32_tab[((BYTE) crcval ^ (*b++)) & 0xff] ^ (crcval
>> 8);
    return ~crcval;
}


/*----------------------------------------------------------------
-------*/
/* update a running CRC with a single byte
        */
/*----------------------------------------------------------------
-------*/
void update_crc( BYTE c, unsigned long *crc )
{
    *crc = crc_32_tab[((BYTE)*crc ^ c) & 0xff] ^ (*crc >> 8);
}
```

```
/*+module definition**********************************************
******
**
** Copyright (c) AlliedSignal Inc.█████
**
**
** Name        : UNCRYPT
**
** Description : This module is an "antidote" for a database file which
 has
**                been encrypted by CRYPTDB.EXE.
**
** Global Procedures:
**   Name                        Type    Abstract
**   ----                        ----    --------
**
** Header for Code Management Software
**
**
*****************************************************************
****-*/
#include <stdio.h>
#include <stdlib.h>
#include <io.h>
#define TAG_SIZE 162

typedef unsigned char BYTE;

void gen_crc_tab( void );
unsigned long get_crc( void *buffer, register int length );
void update_crc( BYTE c, unsigned long *crc );

unsigned long crc_32_tab[256];           /* 32-bit CRC table */

void main( void )
{
    FILE *infp;
    FILE *outfp;
    unsigned long lCRC32;
    long int i, llen;
    BYTE b, cInByte;
    char infname[80];
    char outfname[80];
    long int    db_stamped_key;                  /* database key read fr
om file */

    do
```

```c
        for (i = 0; i < sizeof(p)/sizeof(int); i++)
            e |= 1L << (31 - p[i]);

        /* Compute table of CRC's */
        crc_32_tab[0] = 0x00000000L;
        for (i = 1; i < 256; i++)
        {
            c = 0;
            for (k = i | 256; k != 1; k >>= 1)
            {
                c = c & 1 ? (c >> 1) ^ e : c >> 1;
                if (k & 1)
                    c ^= e;
            }
            crc_32_tab[i] = c;
        }
    }


/*------------------------------------------------------------------
-------*/
/* compute a CRC for a given byte stream
        */
/*------------------------------------------------------------------
-------*/
unsigned long get_crc( void *buffer, register int length )
{
    register unsigned long crcval = 0xffffffffL;
    register BYTE *b = buffer;
    while (length--)
        crcval = crc_32_tab[((BYTE) crcval ^ (*b++)) & 0xff] ^ (crcval
>> 8);
    return ~crcval;
}


/*------------------------------------------------------------------
-------*/
/* update a running CRC with a single byte
        */
/*------------------------------------------------------------------
-------*/
void update_crc( BYTE c, unsigned long *crc )
{
    *crc = crc_32_tab[((BYTE)*crc ^ c) & 0xff] ^ (*crc >> 8);
}
```

Name	Modified		Size	Ratio	Packed	Path
Cryptdb.c	▓▓▓▓▓	1:11 PM	14,581	68%	4,686	
Uncrypt.c	▓▓▓▓▓	10:45 AM	5,451	62%	2,048	
2 file(s)			**20,032**	**66%**	**6,734**	